# Fișă de lucru

# Probleme - Atestat 2012

# Subiectul 1.

Să se realizeze o interfaţă pentru gestionarea persoanelor din evidenţa unei asociaţii de locatari. Se utilizează o bază de date cu o tabela având următoarea structură: ID(N,6), NUME(C,20),NR\_BLOC (C,5), NR\_AP(N,4), ANGAJAT(L). Să se creeze baza de date şi tabela, introducând 10 înregistrări. Interfaţa trebuie să ofere posibilitatea efectuării următoarelor acţiuni:

1. Introducerea de noi persoane în evidenţă,
2. Afişarea tuturor persoanelor care sunt angajate.

Rezolvare în **C#:**

1. Se crează o aplicaţie de tip **WindowsFormsApplication** numită **pb1**.
2. Pentru a crea baza de date, click dreapta pe numele proiectului, **Add →** **New** I**tem** şi alegeţi **Service-based Database**, daţi numele **asociaţie.mdf** şi apăsaţi butonul **Add**. Aşteptaţi puţin .
3. În tab-ul **Server Explorer** **(**apare de obicei unde este şi Toolbox-ul. Dacă nu apare, alegeţi din meniul **View** apoi alegeţi **Server Explorer)**, expandaţi nodul **asociaţie.mdf**, daţi **click dreapta pe Tables** şi alegeţi **Add New Table** pentru a adăuga o nouă tabelă în baza de date. Completaţi structura tabelei după cerinţele problemei. Veţi alege tipurile câmpurilor astfel:

* pentru **ID** – alegeţi tipul **numeric(6,0)** şi din fereastra **Column Properties** alegeţi pentru proprietatea **Identity Specification** valoarea **Yes**, cu **Identity Increment** având valoarea **1** şi **Identity Seed** tot valoarea **1** (astfel calculatorul va completa automat valorile din câmpul ID începând de la 1, valori consecutive, nepermiţând ca în acest câmp să fie două valori identice)
* pentru **NUME** – alegeţi tipul nvarchar, iar din fereastra **Column Properties** alegeţi pentru proprietatea **Length** valoarea **20**
* pentru **NR\_BLOC** – alegeţi tipul **nvarchar**, de lungime **5**.
* pentru **NR\_AP** – alegeţi tipul **int**.
* pentru **ANGAJAT** – alegeţi tipul **bit**, care permite ca valori doar: **true** şi **false**.

1. In fereastra cu instructiuni SQL, inlocuiti numele implicit Table cu numele **locatari** apoi din fereastra de Design a tabelului apasati **Update**.
2. În fereastra **Server Explorer**, daţi **click dreapta pe** tabela **locatari** şi alegeţi opţiunea **Show Table Data.** Adăugaţi 10 înregistrări.
3. Pe forma **Form1**, adăugaţi din **Toolbox** un control de tip **MenuStrip** şi completaţi cu opţiunile: ***Introducerea unei noi persoane, Afisarea celor angajati*** şi ***Ieşire***.
4. În fereastra **Data Sources (**daca nu apare alegeti din meniul **View** optiunea **Other Windows** si apoi alegeti **Data Sources)**, daţi **click pe** butonul **Add New Data Source**, Next, Next, Next, Next, bifati **Tables**, lasati numele implicit, apăsaţi butonul **Finish**.
5. În fereastra **Data Source**, selectaţi tabela **locatari** şi trageţi-o pur şi simplu pe **Form1**. Va apărea automat un control de tip **GridView** în care vor putea fi afişate informaţiile din tabelă.
6. Pentru verificare apăsaţi tasta **F5** sau butonul **Start Debugging**.
7. Pentru a introduce în evidenţă o nouă persoană, se va afişa o nouă formă în a cărei controale vom introduce informaţiile. Aceste controale vor fi legate la tabela **locatari**. Forma va apărea pe ecran în momentul în care vom da click în meniu pe opţiunea ***Introducerea unei noi persoane***. Pentru aceasta parcurgeţi paşii următori:

* Pentru a adăuga priectului o nouă formă, în fereastra **Solution Explorer**, daţi c*lick dreapta pe numele proiectului*, selectaţi **Add** şi apoi **Windows Form**. Lăsaţi în acest exemplu numele implicit, **Form2**.
* În fereastra **Data Sources**, selectaţi tabela **locatari**, deschideţi lista ascunsă din dreptul ei şi selectaţi opţiunea **Details**. Selectaţi apoi tabela şi trageţi-o pur şi simplu pe **Form2**. Vor apărea automat controale de tip textBox şi checkBox (legate la tabela locatari) în care vor putea fi introduse informaţii noi. Pentru controlul **checkBox1** setaţi proprietatea **Text** la şirul vid.
* Reveniţi în **Form1** şi pentru a trata evenimentul **Click** al opţiunii ***Introducerea unei noi persoane***, veţi scrie codul următor:

**Form2 f = new Form2();**

**f.ShowDialog*();*** *// afiseaza Form2, ca dialog modal acest tip de*

*// fereastra dialog nu permite posibilitatea accesarii altor //ferestre ale aplicatiei pana cand acest dialog nu este inchis*

**locatariTableAdapter.Fill(asociatieDataSet.locatari);**

*// se vor reactualiza datele in tabela si implicit si* *in GridView*

*//* *care e legat de aceasta*

1. Pentru a afla toate persoanele *angajate* a căror informaţii sunt reţinute în tabela **locatari**, se va realiza o ***interogare***. Pentru aceasta procedaţi în felul următor:

* În fereastra **Data Sources**, daţi **click dreapta pe** tabela **locatari** şi alegeţi opţiunea **Edit DataSet with Designer.**
* Daţi **click dreapta pe locatariTableAdapter** şi alegeţi opţiunea **Add → Query.** Va apărea fereastra TableAdapter Query Configuration Wizard, în care este selectată implicit opţiunea **Use SQL statements**. Apăsaţi butonul **Next**.
* Va trebui să alegeţi ce comandă SQL vreţi să folosiţi, alegeți **INSERT**. Apăsaţi butonul **Next**.
* Se va afişa implicit o comandă **INSERT** care adaugă o nouă înregistrare în tabela **locatari**. Pentru a o modifica sau testa această comandă apăsaţi butonul **Query Builder…** Comanda folosită este dată mai jos:

**INSERT INTO locatari**

**(ID, NUME, NR\_BLOC, NR\_AP, ANGAJAT)**

**VALUES (@ID,@NUME,@NR\_BLOC,@NR\_AP,@ANGAJAT)**

Apăsaţi butonul **OK,** apăsaţi butonul **Next**. Observaţi că automat s-a generat metoda **InsertQuery** (acum lăsaţi pentru această metodă numele generat implicit, în viitor dacă doriţi puteţi schimba numele cu altul ) .

* Apăsaţi butonul **Next** şi apoi **Finish** pentru a termina interogarea.

Reveniți pe forma **Form2** și adăugați din **Toolbox** un control de tip **Button**. Modificați proprietatea **Name** a acestuia la ***addButton***, iar proprietatea **Text** a acestuia la ***Adăugati*** !. Când veți da click pe acest buton se va adăuga în tabelă o nouă înregistrare cu informațiile completate în formă. Pentru a trata evenimentul **Click** al butonului veți scrie codul de mai jos:

persoaneTableAdapter.InsertQuery(int.Parse(idTextBox.Text), numeTextBox.Text, nr\_BlocTextBox.Text, int.Parse(nr\_ApTextBox.Text), angajatCheckBox.Checked);

persoaneTableAdapter.Update(locatariDataSet); // pentru salvarea datelor

Close(); //închide forma

**Lansaţi în execuţie aplicaţia şi testaţi.**

1. Pentru a afla toate persoanele *angajate* a căror informaţii sunt reţinute în tabela **locatari**, se va realiza o ***interogare***. Pentru aceasta procedaţi în felul următor:

* În fereastra **Data Sources**, daţi **click dreapta pe** tabela **locatari** şi alegeţi opţiunea **Edit DataSet with Designer.**
* Daţi **click dreapta pe locatariTableAdapter** şi alegeţi opţiunea **Add → Query.** Va apărea fereastra TableAdapter Query Configuration Wizard, în care este selectată implicit opţiunea **Use SQL statements**. Apăsaţi butonul **Next**.
* Va trebui să alegeţi ce comandă SQL vreţi să folosiţi. Va fi selectată implicit comanda care ne trebuie: **SELECT wich returns rows**. Apăsaţi butonul **Next**.
* Se va afişa implicit o comandă **SELECT** care întoarce toate înregistrările din tabela **locatari**. Aceasta va trebui modificată. Pentru aceasta, apăsaţi butonul **Query Builder…** şi modificaţi comanda ca mai jos:

**SELECT ID, NUME, NR\_BLOC, NR\_AP, ANGAJAT**

**FROM locatari**

**WHERE (ANGAJAT = 'true')**

apoi apăsaţi butonul **Execute Query** pentru a verifica corectitudinea comenzii. Dacă comanda e bună, apăsaţi butonul **OK,** apăsaţi butonul **Next**. Observaţi că automat vi s-au generat metodele **FillBy** şi **GetDataBy** cu ajutorul cărora veţi face legătura dintre interogarea voastră şi controalele în care afişaţi rezultatul interogării (acum lăsaţi pentru aceste metode numele generate implicit, în viitor dacă doriţi puteţi schimba numele cu altele sugestive, de ex. FillByAngajati, respectiv GetDataByAngajati) .

* Apăsaţi butonul **Next** şi apoi **Finish** pentru a termina interogarea.
* Pe **Form1**, daţi dublu click în **MenuStrip** pe opţiunea ***Afisarea celor angajati*** pentru a trata evenimentul **Click** şi scrieţi codul următor pentru a afişa în controlul de tip GridView doar persoanele angajate. Veţi afişa de fapt rezultatul interogării realizate anterior.

**locatariTableAdapter.FillBy(asociatieDataSet.locatari);**

1. Pentru a putea reafisa toti locatarii (dupa ce am ales de exemplu sa-i afisam doar pe cei angajati):

* pe **Form1** adăugaţi din **Toolbox** un control de tip **Button**. Setaţi atât proprietatea **Name** cât şi proprietatea **Text** la **Refresh**. Pentru tratarea evenimentului **Click** al acestui buton scrieţi codul următor care va reactualiza datele despre animale în controlul de tip DataGridView:

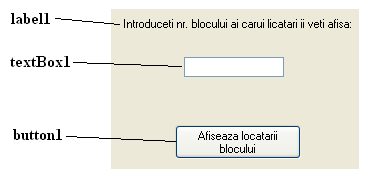
this.locatariTableAdapter.Fill(this.asociatieDataSet.locatari);

**Lansaţi în execuţie aplicaţia şi testaţi.**

**ALTE CERINTE**

Pentru exerciţiu, în plus faţă de cerinţele problemei, vom adăuga în meniu şi opţiunea: ***Afisarea locatarilor unui bloc dat***.

Adăugaţi în proiect o nouă formă denumită **Form3**. Pe aceasta adăugaţi din **ToolBox** trei controale ca în figură:



În fereastra **Data Sources**, selectaţi tabela **locatari**, deschideţi lista ascunsă din dreptul ei şi selectaţi opţiunea **DataGridView** şi apoi trageţi-o pur şi simplu pe **Form3**. Va apărea automat un control de tip **GridView** în care vor putea fi afişate informaţiile din tabelă.

În clasa **Form3** adăugaţi o dată membră publică de tip *string* numită **bloc** astfel:

**public string bloc;**

În continuare realizaţi pe rând următorii paşi (aceeaşi ca la crearea interogării anterioare doar că se va schimba comanda şi numele metodelor generate la final):

* În fereastra **Data Sources**, daţi **click dreapta pe** tabela **locatari** şi alegeţi opţiunea **Edit DataSet with Designer.**
* Daţi **click dreapta pe locatariTableAdapter** şi alegeţi opţiunea **Add → Query.** Va apărea fereastra TableAdapter Query Configuration Wizard, în care este selectată implicit opţiunea **Use SQL statements**. Apăsaţi butonul **Next**.
* Va trebui să alegeţi ce comandă SQL vreţi să folosiţi. Va fi selectată implicit comanda care ne trebuie: **SELECT wich returns rows**. Apăsaţi butonul **Next**.
* Se va afişa implicit o comandă **SELECT** care întoarce toate înregistrările din tabela **locatari**. Aceasta va trebui modificată. Pentru aceasta, apăsaţi butonul **Query Builder…** şi modificaţi comanda ca mai jos:

**SELECT ID, NUME, NR\_AP, ANGAJAT**

**FROM locatari**

**WHERE ( NR\_BLOC = @bloc )**

apoi apăsaţi butonul **Execute Query** pentru a verifica corectitudinea comenzii.

***Observaţi că pentru a folosi valoarea unei date membre a unei forme se scrie în faţa ei @ (, iar data membră folosită, trebuie să fie publică) .***

Dacă comanda a fost verificată şi este bună, apăsaţi butonul **OK,** apăsaţi butonul **Next**. Observaţi că automat vi s-au generat metodele **FillBy1** şi **GetDataBy1.** Redenumiţi-le cunume sugestive, de ex. **FillByBloc,** respectiv **GetDataByBloc** .

* Apăsaţi butonul **Next** şi apoi **Finish** pentru a termina interogarea.
* Pe **Form1**, daţi dublu click în **MenuStrip** pe opţiunea ***Afisarea locatarilor unui bloc dat*** pentru a trata evenimentul **Click** şi scrieţi codul următor pentru a afişa **Form3** (în care veţi introduce numărul blocului şi veţi a afişa într-un control de tip GridView doar persoanele din blocul dat. Veţi afişa de fapt rezultatul interogării realizate anterior)

**Form3 f3 = new Form3();**

**f3.ShowDialog();**

În **Form3**, în handler-ul de tratare a evenimentului **Click** pentru **button1** scrieţi următorul cod care salvează valoarea introdusă în **textBox1** în data membră **bloc** şi care afişează rezultatul interogării realizate anterior:

**bloc = int.Parse(textBox1.Text);**

**locatariTableAdapter.FillByBloc(asociatieDataSet.locatari,bloc);**

**Rulaţi şi testaţi aplicaţia.**